CS 134: Networks	Lecture 13, Monday, 3/27
Prof. Yaron Singer	Spring 2017

1 Overview

In today's lecture, we discuss threshold models, and in particular, show that influence under a threshold model is also monotone submodular by drawing connections between threshold model influence and a slightly modified version of cascade influence in a model known as correlated random graphs.

2 Recap: Submodularity

We conduct a quick review of submodularity.

Definition. *Given a ground set of elements $V = \{a_1, \cdots, a_n\}$, a function $f : 2^{|V|} \Rightarrow \mathbb{R}$ that takes in some subset of V and returns a value in $\mathbb{R}$ is submodular on V if for all subsets $S, T \subseteq V$ s.t. $S \subseteq T$ and for all $a \notin T$:*

$$f(S \cup \{a\}) - f(S) \geq f(T \cup \{a\}) - f(T)$$

We denote $f(S \cup \{a\}) - f(S)$ as $f_S(a)$, and thus the above condition becomes simply $f_S(a) \geq f_T(a)$. Note that $f_S(a)$ is also known as the marginal contribution of a to S under f.

Intuitively, this means that when a is added to a subset S, some property (as measured by f) of S increases more than does the same property of T when we add a to T, given that S is a subset of T. Some such functions are additive functions and coverage functions.

2.1 Examples of Submodular Functions

Additive functions. For additive functions, we assign every element some weight and define $f(S)$ to be the sum of the weights of the elements in S, i.e.:

$$f(S) = \sum_{a \in S} w_a$$

In additive functions, for all S and a,

$$f_S(a) = \sum_{i \in (S \cup \{a\})} w_i - \sum_{i \in S} w_i \tag{1}$$

$$= \left(\sum_{i \in S} w_i + w_a \right) - \sum_{i \in S} w_i \tag{2}$$

$$= w_a \tag{3}$$

Thus, the marginal contribution of an element a, no matter what S is, is w_a, and $f_S(a) = f_T(a)$, which certainly satisfies the submodularity condition that $f_S(a) \geq f_T(a)$.

Coverage functions. For coverage functions, the ground set of elements are "circles" that cover "points", i.e. sets that consist of points (much like the setup of the famous vertex cover problem):

$$V = \{C_1, \cdots, C_n\}$$

The value $f(S)$ is defined as the number of points shared among all "circles" in S, i.e.:

$$f(S) = |\cup_{C_i \in S} C_i|$$

The marginal contribution of an element a is therefore the number of points in a that are not in the union of S. The submodularity condition $f_S(a) \geq f_T(a)$ is easy to see: when you add more "circles" to a set of "circles", the number of points uniquely contributed by another "circle" a can only decrease.

2.2 Monotone Submodularity and Greedy Approximation

The submodularity condition gives rise to a nice theorem:

Theorem 1. *For any monotone submodular function f, the greedy algorithm returns a set S s.t.:*

$$f(S) \geq \left(1 - \frac{1}{e}\right) \texttt{OPT}$$

where `OPT` *is* $\max_{T:T\subset V,|T|\leq k} f(T)$.

Essentially, the greedy algorithm returns a set S that comes within approximately 63% of maximizing f under the constraint that the size of S is less than or equal to k. The proof of this has been covered in the previous lecture.

3 Recap: Independent Cascades

We begin with a graph $G = (V, E, P)$, where P is a vector of weights or probabilities, $P \in [0, 1]^{|E|}$.

At time step $t = 0$, infect the initial set of nodes S. At every time step $t \geq 1$ thereafter, any nodes that were infected at the previous time step $t-1$ will attempt to infect any healthy neighbor nodes, succeeding with probability equal to the weight of the edge between them. Edges can thus transmit *infections'*, and when they do so, this is called *realization* of that edge.

The influence function f on S is the expected number of nodes that are infected after n time steps, when starting with the infected set S at $t = 0$. Influence maximization concerns finding a set of size k that maximizes this expectation. This function has been shown to be submodular in previous lectures.

You may think of edges that are realized as "live" edges, and those that aren't as "blocked" edges. $f(S)$ is then the expected value of the number of nodes reachable via live edges. Note that any edge only ever has at most one chance at being realized - we can then use the definition of expectation and the probabilities of each possible set of realizations of edges to calculate $f(S)$ as:

$$f(S) = \sum_{G_i} \mathbb{P}(G_i) \cdot |\text{nodes reachable from } S \text{ in } G_i|$$

This can be thought of as a coverage function. For any given G_i, the nodes reachable from S in G_i is just the value of a coverage function! For each node in S, construct the set of nodes reachable from it - $f(S)$ is then just the union of those sets, and $f(S \cup \{a\})$ is just the union of those sets and the nodes reachable from a.

Each individual G_i then composes one monotone submodular function, and the original $f(S)$ is then just a weighted sum of monotone submodular functions, which itself is also a monotone submodular function.

The greedy solution is however very computationally expensive to compute, as you need to enumerate an exponential number of graphs to take the next element with largest marginal contribution. This problem has Sharp-P complexity, in fact. However, one may optimize an approximation of this.

4 Threshold Models

We now consider the main topic of today's lecture. In essence, every edge is assigned some weight, and any given node is infected based on whether the sum of the weights of edges connected to infected nodes exceeds some threshold. This threshold is intuitively your probability of being influenced, and the edge weights can be interpreted as the node's susceptibility to infection by another node.

The linear threshold model. More formally, in the linear threshold model we are given a directed graph $G = (V, E, \mathbf{w})$, where $\mathbf{w} \in [0, 1]^{|E|}$ such that the sum of the weights of incoming edges for any node never goes above 1, i.e. $\sum_{u \in N(v)} w_{v,u} \leq 1$ for any node v, where $N(v) = \{u \in V : (u, v) \in E\}$ is the set of nodes with edges coming into v, otherwise known as the in-neighborhood. Moreover, every node v has a threshold θ_v, chosen uniformly at random from $[0, 1]$. We begin with some infected nodes S at time step $t = 0$. At each time step, every healthy node v such that $\sum_{u \in \{\text{all thus far infected nodes}\}} w_{u,v} \geq \theta_v$ becomes infected.

There are two major changes from the independent cascades model here. Firstly, every node has only a single chance to infect its neighbors in the independent cascades model, whereas here infected nodes continue contributing "infective" influence. Secondly, we are no longer looking repeatedly at random chances of being infected, but a single randomly chosen threshold of infection that's assigned to a node.

We now propose the following theorem:

Theorem 2. *Influence in the Linear Threshold model is a monotone submodular function.*

Proof. As customary, we define $f(S)$ to be the expected number of nodes infected when S is selected at $t = 0$ in the Linear Threshold model. To prove that f is monotone submodular, we construct another function f' and model that looks very similar to the independent cascades process, and show that f' achieves the same effect. We define the Correlated Random Graph model:

> **Definition.** *A directed graph $G = (V, E, \boldsymbol{w})$ is modeled as a Correlated Random Graph if every node v has at most **one** incoming edge connected to it, where each edge (u, v) from node u to v realizes with probability $w_{u,v}$ and no edges connect to v with probability $1 - \sum_{u \in V} w_{u,v}$. Note that $\sum_{u \in V} w_{u,v} \leq 1$ for every node v.*

We let $f(S)$ be again the expected number of nodes reachable from set S at time step $t = 0$ in graph G. Note that f above is monotone submodular, for the same reasons as in the independent cascades model.

We now claim that influence in the Linear Threshold model is exactly the same as influence in the CRG model. Let $I_0, I_1, \cdots, I_n$ be the nodes that are influence or activated at each time step, with $I_0 = S$ in the LT model. We show that the probability of a node being activated at some time step t in the LT model is the same as its probability of being activated in the CRG model given the same starting set S using induction.

Base case (I_0): trivial, either a node is in S or it isn't, and S is the same for both models.

Inductive step: Given that the expected distributions of I_{t-1} are the same in both models, we need to show that the probability of any node v being activated at time step t is the same, thus showing that distribution I_t is the same for both models.

To see this, consider the LT model. Note that the threshold θ_t is chosen uniformly at random, but since it wasn't activated at $t-1$, the threshold is at least the sum of the weights of connected edges to node that were activated during time step $t-1$, i.e. it's at least $\sum_{u \in I_{t-1}} w_{u,v}$. For it to be activated during time step t, the threshold would have to be less than $\sum_{u \in I_t} w_{u,v}$, and thus its probability of being activated in the LT model at time step t is:

$$P_t(v \text{ activated}) = \frac{\sum_{u \in I_t \setminus I_{t-1}} w_{u,v}}{1 - \sum_{u \in I_{t-1}} w_{u,v}}$$

Similarly, in the CRG model, the probability that the incoming edge if any into a node v does not come from I_{t-1} is $1 - \sum_{u \in I_{t-1}} w_{u,v}$, but does come from I_t is $\sum_{u \in I_t / I_{t-1}}$, so the conditional probability of v being activated in the CRG model at time step t is the same as above.

Thus I_t is distributed the same under both models, and the proof is complete. □